Remarks

The various parts of the Office Action (and other matters, if any) are discussed below under appropriate headings.

The Previously Added Limitations

The Examiner states that the previously added limitations pertaining to the content of the recovered program are merely non-functional descriptive language, and specifically states that there is no language that functionally links the newly added language to the system, method or computer readable medium.

In response thereto, Applicants have amended the independent claims to recite that the recovered program *is called by the microprocessor*, thereby functionally linking the previously added language to the program, method, and device claims.

In view of the above, it is respectfully submitted that the previously added limitations are to be given patentable weight.

Claim Rejections - 35 USC § 103

Claims 1, 3 and 6-9 stand rejected under 35 USC §103(a) based on U.S. Patent No. 5,982,887 (*Hirotani*) in view of U.S. Patent No. 6,907,125 (*Oishi*) in further view of Applied Cryptography (*Schneier*), in further view of U.S. Patent No.6,526,462 (*Elabd*), and in further view of Navigating C++ and Object-Oriented Design (*Anderson*). Further, claims 1, 3 and 6-9 also stand rejected under 35 USC §103(a) based on *Hirotani, Schneier*, *Elabd* and *Anderson* in view of U.S. Patent No. 5,613,005 (*Murakami*). Withdrawal of the rejections is respectfully requested for at least the following reasons.

Anderson Does Not Make up for the Deficiencies of Hirotani, Oishi, Schneier, Elabd and Murakami

Independent claims 1, 3, 6 and 8 recite a recovered program that includes at least a public function which is to be called from outside of the recovered program by a microprocessor and an internal function which is to be called from inside of the recovered program, and a relative address list indicating a relative address of the at least one public function in the recovered program, wherein the relative address list is provided at a prescribed location in the recovered program.

For example, and with reference to Fig. 6 of the present application, there is shown a structure of a recovered program 503. The recovered program 503 includes a relative address list 60 and a program portion 66. The program portion includes public functions 61-62 that are called from the outside of the recovered program 503, and internal functions 63-65 that are called from the inside the recovered program 503 using relative addresses. The relative address list 60 includes the relative addresses of the public functions 61-62 (see pg. 11, ln. 32 - pg. 12, ln. 26 of the application as filed). The microprocessor unit 102 can call the public functions 61-62 in the recovered program 503 (see pg. 13, lns. 11-17).

The Examiner relies on *Anderson* for teaching that object-oriented designs include a public function which is to be called from outside of the recovered program and an internal function which is to be called from inside the recovered program, and a relative address list indicating a relative address of the at least one public function in the recovered program, wherein the relative address list is provided at a prescribed location in the program. Applicants respectfully disagree with the Examiner for at least the following reasons.

In Chapter 4 of *Anderson* (cited by the Examiner), *Anderson* simply discusses the topic of "Encapsulation" in C++ programming language. Encapsulation, as known to one skilled in the art, is simply the grouping together of data and functionality. C++ implements encapsulation by allowing all members of a class to be declared as either public, private or protected.

In particular, *Anderson* discusses how to design a Fifo (First In, First Out) data type throughout substantially the entirety of Chapter 4. On pages 175-176, which are specifically referred to by the Examiner, *Anderson* simply discusses the capability of C++ that allows the Fifo data type to be designed with private and public sections. The design of a Fifo data type with private and public sections, however, does not make obvious the limitation of the independent claims, which require *a recovered program recovered from a concealed program* to include a public function, an internal function and a relative address list.

The Examiner also refers to pages 92-93 (Chapter 3) of *Anderson* as teaching a "relative address list". This section of *Anderson*, however, has not been found to teach or suggest that the relative address list should be included *in the recovered program*

recovered from a concealed program and provided at a prescribed location in the recovered program. The mere fact that *Anderson* describes "pointers" does not lead to the use a relative address list in a *recovered program*. Moreover, *Anderson* does not disclose a relative address list for indicating a relative address of at least one public function in the recovered program.

Thus, since *Anderson* has not been found to teach or suggest the above limitations of the independent claims, the obviousness rejection should be withdrawn.

Hirotani Is Not Properly Combinable with Schneier

As admitted by the Examiner, *Hirotani* clearly fails to disclose that the decryption is performed by a hardware circuit. In an attempt to make up for such deficiency, the Examiner refers to *Schneier* and alleges that the combination of *Hirotani* and *Schneier* would result in the use of a hardware decryption circuit in place of the software decryption used by *Hirotani*.

The purpose of the invention as disclosed by *Hirotani* is to provide an encrypted program executing apparatus that performs decryption by a software method that overcomes problems associated with conventional software methods. In particular, *Hirotani* discloses that a conventional system implementing a software method has drawbacks in which a decrypting program (i.e., program used to decrypt the encrypted program) itself is copied (see col. 1, lns. 20-29).

Accordingly, the objective of the invention as disclosed by *Hirotani* is to overcome such drawbacks by providing a software method that ensures that the decryption program cannot be read out of the microcomputer 10, thus preventing the algorithm of the decrypting program from being analyzed by a third party (see col. 3, lines 40-43 and col. 4, lines 1-4 and 28-51).

It is respectfully submitted that it is not clear if it is even possible to implement a hardware solution to the software system of *Hirotani*. Moreover, even if possible, it is not clear how one would go about modifying the system of *Hirotani* so as to implement a hardware solution that performs all of the features of the software solution.

Further, even if it is possible to implement a hardware solution to the system of *Hirotani*, such hardware implementation would destroy the principle of operation of *Hirotani*.

According to § 2143.01(VI) of the MPEP, **THE PROPOSED MODIFICATION CANNOT CHANGE THE PRINCIPLE OF OPERATION OF A REFERENCE**. If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. In re Ratti, 270 F.2d 810, 123 USPQ 349 (CCFA 1959).

In this case, modifying *Hirotani's* invention with hardware decryption instead of software decryption would essentially change the principle of operation of *Hirotani*. In particular, the sole-purpose of the invention as disclosed by *Hirotani* (e.g., see Figures 1 and 3) is to specifically design an encrypted program executing apparatus that provides software decryption and at the same time, does not have the drawbacks of conventional software decryption as discussed in the background of *Hirotani*. The Examiner's proposed modification of *Hirotani* to hardware decryption clearly changes the principle of operation of *Hirotani* and, therefore, is improper.

Moreover, *Schneier* teaches hardware encryption to encrypt telephone conversations, facsimile transmissions and so on. With respect to microcomputers, *Schneier* specifically states that *[i]t is cheaper to put special-purpose encryption hardware in the telephones, facsimile machines and modems than it is to put in a microprocessor*... (see pg. 224, Ins. 32-36). Thus, *Schneier* is effectively teaching away from using hardware encryption with microprocessors.

For at least the above reasons, the combination of *Hirotani* with *Schneier* is improper and, thus, the Examiner has not established a *prima facie* case of obviousness.

Oishi in View of Elabd Does Not Teach a Data Scramble Circuit That Is Single Hardware Circuit

As presented in the reply to the Office Action of February 14, 2006 and again in the reply to the Office Action dated July 20, 2006, Applicants maintain that *Oishi* and/or

Elabd, alone or in combination, do not teach a data scramble circuit that is a single hardware circuit, and at least a portion of the data scramble circuit is operative to perform both a data scramble function and an error correction function.

Oishi clearly teaches two separate and distinct circuits for performing decryption and error correction. Thus, *Oishi* fails to teach a data scramble circuit that is a single hardware circuit. Further, while *Elabd* teaches a "system on chip" (SOC) design, such SOC design, in view of *Oishi*, yields a single chip that includes a decryption circuit and a separate and distinct error correction circuit. The fact that such circuits may be on the same substrate does not overcome the fact that they are still separate circuits.

Murakami does not Teach a Data Scramble Circuit that is Operable to Perform Both a Data Scramble Function and an Error Correction Function

The Examiner implies that the claims are rendered obvious even when *Oishi* is replaced by *Murakami*. Applicants respectfully disagree for at least the following reasons.

Murakami discloses an encryption circuit and a decoding circuit that is immune to the effects of missing data after 12 bits (see col. 6 lns. 18-26 of *Murakami*). In other words, if the data within a received data stream is missing or otherwise not present, then the result of that data string will not be affected by this missing data. There is no teaching in *Murakami*, however, that error correction is performed on the data by the encryption circuit or the decoding circuit. For example, the received data may be complete (i.e., no missing data), but the data contained therein may be wrong (e.g., a bit of the received data may be different from the corresponding bit of the transmitted data). *Murakami* does not teach that such data can be corrected.

Accordingly, the cited art has not been found to teach or suggest all the features of claims 1, 3, 6 and 8. Accordingly, withdrawal of the rejection of claims 1, 3, 6 and 8 is respectfully requested.

Claims 7 and 9 depend from claims 6 and 8, respectively, and therefore can be distinguished from the cited art for at least the same reasons.

Accordingly, withdrawal of the rejection of claims 7 and 9 is respectfully requested.

Conclusion

In view of the foregoing, request is made for timely issuance of a notice of allowance.

Respectfully submitted,

RENNER, OTTO, BOISSELLE & SKLAR, LLP

By /Kenneth W. Fafrak/
Kenneth W. Fafrak, Reg. No. 50,689

1621 Euclid Avenue
Nineteenth Floor
Cleveland, Ohio 44115
(216) 621-1113

CERTIFICATE OF MAILING OR FACSIMILE TRANSMISSION UNDER 37 CFR 1.8(a)

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is
___ being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Commissioner for Patents address below.
__ being transmitted via EFS or via facsimile to **(571) 273-8300 (Centralized Facsimile Number)** at the U.S. Patent and Trademark Office.
X submitted on the date shown below using the U.S. Patent Office's Electronic Filing System.

/Kenneth W. Fafrak/ — **September 21, 2007**
Kenneth W. Fafrak — Date

R:\Ken\Y\YAMA\P0748US\Reply_to_OA_6_22_07.wpd